**Book Store with Entity Manager**

**Objective:**

Create basic bookstore app using Asynchronous Query Handler and Entity Manager.

**Description:**

In this app I have used basic activities and xml file to create an interaction and user interface. In addition to contract and database packages from previous assignment, I have created content providers to design app more efficiently and make it available to other applications.

I have created a content provider that can be used to uniquely identify rows and tables from the database table. I have implemented Loader manager in order to enable the app to work in background rather than making it run on the main thread. I have created a custom cursor adapter that is integrate with the loader manager to display the books and authors and run operations asynchronously. I have used Context action bar in order to allow deletion operation. For SQLite database I have created adapter class which includes inner class that is used to create and update database.

In addition to previous application this application includes entity manager and asynchronous query handler callbacks along with several interfaces that runs all the CRUD operations in the background. This makes the application more efficient and does no block the main activity. Also contract class is used to access underline database structure through book and author class. In This app I have created 4 activities that interact with each other in order to carry out desired tasks. I have also defined a custom adapter to display the list of books in main activity. Also it contains entities that are used to create objects which implements parcelable interface so that they can be passed from one activity to other through intents short messages. I used number of inbuilt methods to provide functionality. Used joint query is used to display data related to book and author.

**Conclusion:**

In this way I have successfully created Book Store App that can store, display books using Entity Manager and Asynchronous Query Handler.